

Algoritmos de pesquisa sequencial e binária

1. Contextualização

1.1 Pesquisa sequencial

É uma das buscas mais simples pois percorre registro por registro em busca da chave digitada pelo usuário.

Sua complexidade é dada por:

- Se o registro for o primeiro: 1 comparação.
- Se o registro procurado for o último: N comparações.
- Se for igualmente provável que o argumento apareça em qualquer posição da tabela, em média: $(n+1)/2$ comparações.
- Se a busca for mal sucedida: N comparações.
- Logo, a busca sequencial, no pior caso, é $O(n)$.

Caso a busca sequencial seja feita em tabelas ordenadas, no caso médio $N/2$ comparações são feitas se as chaves estiverem ordenadas, pois se para a busca assim que uma chave maior que a procurada é encontrada.

1.1.1 Vantagens

Os itens na tabela poderão ser examinados sequencialmente sem que todos os registros precisem ser acessados: o tempo de busca diminui consideravelmente.

1.1.2 Desvantagens

A tabela tem que estar ordenada. Exige espaço adicional para armazenar as tabelas de índices.

1.2 Pesquisa binária

O elemento buscado é comparado ao elemento do meio do arranjo:

- Se igual, busca bem-sucedida.
- Se menor, busca-se na metade inferior do arranjo.
- Se maior, busca-se na metade superior do arranjo.

Sua complexidade é $O(\log(n))$, pois cada comparação reduz o número de possíveis candidatos por um fator de 2.

1.2.1 Vantagens

As vantagens da pesquisa binária são a eficiência na busca e a simplicidade de implementação.

1.2.2 Desvantagens

Nem todo arranjo está ordenado, exige o uso de um arranjo para armazenar os dados, ou seja, faz uso do fato que os índices do vetor são inteiros consecutivos, além da inserção e remoção de elementos ser ineficientes.

2. Comparação de tempos

2.1 Para descobrir a posição do primeiro termo em uma sequência ordenada.

Pesquisa sequencial	Pesquisa binária
3 μ s	5 μ s
2 μ s	2 μ s
4 μ s	5 μ s

Média de tempo pesquisa sequencial = 3 μ s.

Média de tempo pesquisa binária = 4 μ s.

2.2 Para descobrir a posição do elemento do meio em uma sequência ordenada.

Pesquisa sequencial	Pesquisa binária
2552 μ s	4 μ s
2547 μ s	4 μ s
2392 μ s	4 μ s

Média de tempo pesquisa sequencial = 2506 μ s.

Média de tempo pesquisa binária = 4 μ s.

2.3 Para descobrir a posição do último elemento em uma sequência ordenada.

Pesquisa sequencial	Pesquisa binária
5045 μ s	5 μ s
5052 μ s	5 μ s
5062 μ s	5 μ s

Média de tempo pesquisa sequencial = 5053 μ s.

Média de tempo pesquisa binária = 5 μ s.

3 Conclusões

Os dois algoritmos são bastante eficientes para arranjos ordenados, porém a pesquisa binária realiza com menor duração de tempo a busca da posição do último elemento e o elemento da metade do vetor. Isso acontece pois em seu algoritmo de

quebra do vetor, ele sempre irá pegar o valor do meio para saber se o número se encontra na primeira parte do vetor (caso seja um número menor que a metade) ou na parte restante (caso seja um número maior que a metade). Em relação a busca pelo último elemento, a pesquisa sequencial como vai buscando registro por registro a partir do primeiro, acaba demorando mais do que a binária que vai dividindo pela metade sempre. Para a busca do primeiro elemento, a pesquisa sequencial realiza a operação mais rapidamente pois como vai do primeiro registro até o ultimo em ordem, ao achar o primeiro registro já sai da função e retorna para a main a posição do resultado digitado pelo usuário.